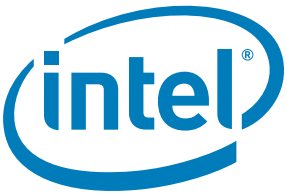


Dual-Core Intel® Xeon® Processor 5000 Series

Specification Update

August 2006

Document Number: 313065-002

Contents

Revision History

Version	Description	Date
001	Initial release of Specification Update	May 2006
002	Added Notes 2-7 to Table 1 Updated branding of Intel(R) 64	August 2006

Preface

This document is an update to the specifications contained in the Affected Documents and Related Documents tables below. This document is a compilation of device and documentation errata, specification clarifications and changes. It is intended for hardware system manufacturers and software developers of applications, operating systems, or tools.

Information types defined in Nomenclature are consolidated into the specification update and are no longer published in other documents.

This document may also contain information that was not previously published.

Affected Documents

1. *Dual-Core Intel® Xeon® Processor 5000 Series Datasheet* (Document Number 313079)

Related Documents

1. *IA-32 Intel® Architecture Software Developer's Manual*, Volume 1: Basic Architecture
   (Document Number 253665)
   Link: ftp://download.intel.com/design/Pentium4/manuals/253665.htm
2. *IA-32 Intel® Architecture Software Developer's Manual*, Volume 2A: Instruction Set Reference, A-M (Document Number 253666)
   Link: ftp://download.intel.com/design/Pentium4/manuals/253666.htm
3. *IA-32 Intel® Architecture Software Developer's Manual*, Volume 2B: Instruction Set Reference, N-Z (Document Number 253667)
   Link: ftp://download.intel.com/design/Pentium4/manuals/253667.htm
4. *IA-32 Intel® Architecture Software Developer's Manual*, Volume 3A: System Programming Guide (Document Number 253668)
   Link: ftp://download.intel.com/design/Pentium4/manuals/253668.htm
5. *IA-32 Intel® Architecture Software Developer's Manual*, Volume 3B: System Programming Guide (Document Number 253669)
   Link: ftp://download.intel.com/design/Pentium4/manuals/253669.htm

Nomenclature

Errata are design defects or errors. These may cause the processor's behavior to deviate from published specifications. Hardware and software designed to be used with any given stepping must assume that all errata documented for that stepping are present on all devices.

S-Spec Number is a five digit code used to identify products. Products are differentiated by their unique characteristics, for example, core speed, L3 cache size, package type, and so forth, as described in the processor identification information table. Read all notes associated with each S-Spec number.

Specification Changes are modifications to the current published specifications. These changes will be incorporated in any new release of the specification.

Specification Clarifications describe a specification in greater detail or further highlight a specification's impact to a complex design situation. These clarifications will be incorporated in any new release of the specification.

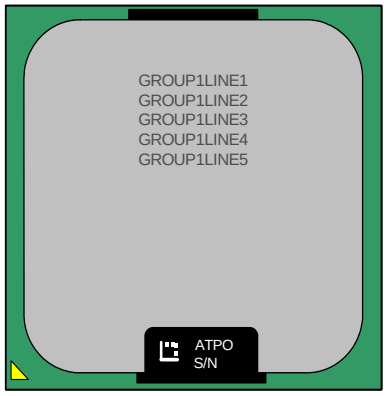
Documentation Changes include typos, errors, or omissions from the current published specifications. These will be incorporated in any new release of the specification.

Note: Errata remain in the specification update throughout the product's lifecycle, or until a particular stepping is no longer commercially available. Under these circumstances, errata removed from the specification update are archived and available upon request. Specification changes, specification clarifications and documentation changes are removed from the specification update when the appropriate changes are made to the appropriate product specification or user documentation (datasheets, manuals, and so forth.).

Package Markings

Dual-Core Intel® Xeon® Processor 5000 Series Package Markings

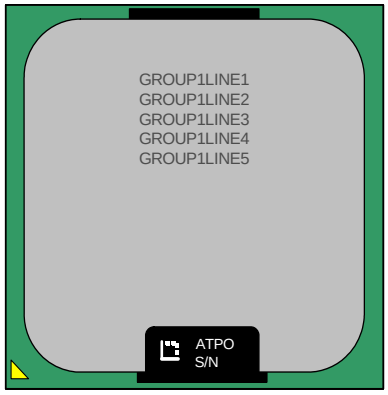
Figure 1. Dual-Core Intel® Xeon® Processor 5000 Series Top-Side Markings (Example)



Legend:	Mark Text (Production Mark):
GROUP1LINE1	3733DP/4M/1066
GROUP1LINE2	Intel ® Xeon ®
GROUP1LINE3	5080 SXXX COO
GROUP1LINE4	i (M) © '05
GROUP1LINE5	FPO

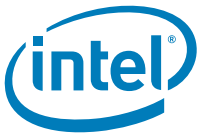
Notes:
1. All characters will be in upper case.
2. Drawing is not to scale.

Figure 2. Dual-Core Intel® Xeon® Processor 5063 (MV) Top-Side Markings (Example)



Legend:	Mark Text (Production Mark):
GROUP1LINE1	3200DP/4M/1066/MV
GROUP1LINE2	Intel ® Xeon ®
GROUP1LINE3	5063 SXXX COO
GROUP1LINE4	i (M) © '05
GROUP1LINE5	FPO

Notes:
1. All characters will be in upper case.
2. Drawing is not to scale.



Identification Information

The Dual-Core Intel Xeon Processor 5000 series can be identified by the following register contents:

Family[1]	Model[2]
1111b	0110b

Notes:
1. The Family corresponds to bits [11:8] of the EDX register after RESET, bits [11:8] of the EAX register after the CPUID instruction is executed with a 1 in the EAX register, and the generation field of the Device ID register accessible through Boundary Scan.
2. The Model corresponds to bits [7:4] of the EDX register after RESET, bits [7:4] of the EAX register after the CPUID instruction is executed with a 1 in the EAX register, and the model field of the Device ID register accessible through Boundary Scan.

Cache and TLB descriptor parameters are provided in the EAX, EBX, ECX and EDX registers after the CPUID instruction is executed with a 2 in the EAX register. Please refer to the *AP-485 Intel® Processor Identification and the CPUID Instruction* Application Note for further information on the CPUID instruction.

Table 1. Dual-Core Intel Xeon Processor 5000 Series Identification Information

S-Spec	Core Stepping	L2 Cache Size (bytes)	CPUID	Core Freq (GHz)	Data Bus Freq (MHz)	Package and Revision	Processor Number	Note
SL968	C-1	2M x 2	F64	3.73	1066	771-pin micro-LGA with 37.55 x 37.55 FCLGA6 Package	5080	2,3,4, 5,6
SL96A	C-1	2M x 2	F64	3.20	1066	771-pin micro-LGA with 37.55 x 37.55 FCLGA6 Package	5060	3,4,5, 6
SL96B	C-1	2M x 2	F64	3.20	1066	771-pin micro-LGA with 37.55 x 37.55 FCLGA6 Package	5063	1,3,4, 5,6
SL96C	C-1	2M x 2	F64	3	667	771-pin micro-LGA with 37.55 x 37.55 FCLGA6 Package	5050	1,2,3, 4,5,6
SL96E	C-1	2M x 2	F64	2.67	667	771-pin micro-LGA with 37.55 x 37.55 FCLGA6 Package	5030	1,2,3, 4,5,6

Notes:
1. These parts are MV (mid-voltage) processors.
2. These parts are enabled for Enhanced Intel SpeedStep® Technology (EIST).
3. These parts are enabled for Enhanced Halt State (C1E).
4. These parts have Execute Disable bit functionality.
5. These parts have Intel(R) Virtualization Technology (VT) enabled.
6. These parts have Hyper-Threading Technology enabled.

Summary Tables of Changes

The following table indicates the Errata, Specification Changes, Specification Clarifications, or Documentation Changes which apply to the Dual-Core Intel® Xeon® Processor 5000 Series. Intel may fix some of the errata in a future stepping of the component, and account for the other outstanding issues through documentation or specification changes as noted. This table uses the following notations:

Codes Used in Summary Table

X: Erratum, Specification Change or Clarification that applies to the given processor stepping.

(No mark) or (Blank Box): This erratum is fixed in listed stepping or specification change does not apply to listed stepping.

Doc: Document change or update that will be implemented.

Plan Fix: This erratum may be fixed in a future stepping of the product.

Fixed: This erratum has been previously fixed.

No Fix: There are no plans to fix this erratum.

Change bar to left of table row indicates this item is either new or modified from the previous version of this document.

Each Specification Update item will be prefixed with a capital letter to distinguish the product. The key below details the letters that are used in Intel's microprocessor Specification Updates:

A = Intel® Pentium® II processor
B = Mobile Intel® Pentium® II processor
C = Intel® Celeron® processor
D = Dual-Core Intel® Xeon® processor 2.80 GHz
E = Intel® Pentium® III processor
F = Intel® Pentium® processor Extreme Edition and Intel® Pentium® D processor
G = Intel® Pentium® III Xeon® processor
H = Mobile Intel® Celeron® processor at 466/433/400/366/333/300 and 266 MHz
I = Dual-Core Intel® Xeon® processor 5000 series
J = 64-bit Intel® Xeon® processor MP with 1 MB L2 cache
K = Mobile Intel® Pentium® III processor
L = Intel® Celeron® D processor
M = Mobile Intel® Celeron® processor
N = Intel ® Pentium® 4 processor
O = Intel® Xeon® processor MP
P = Intel® Xeon® processor
Q = Mobile Intel® Pentium® 4 processor supporting Hyper-Threading Technology on 90 nm process technology
R = Intel® Pentium® 4 processor on 90 nm process
S = 64-bit Intel® Xeon® processor with 800 MHz system bus (1 MB and 2 MB L2 cache versions)
T = Mobile Intel® Pentium® 4 processor-M
U = 64-bit Intel® Xeon® processor MP with up to 8 MB L3 cache
V = Mobile Intel® Celeron® processor on .13 micron process in micro-FCPGA package

W = Intel® Celeron® M processor
X = Intel® Pentium® M processor on 90 nm process with 2 MB L2 cache
Y = Intel® Pentium® M processor
Z = Mobile Intel® Pentium® 4 processor with 533 MHz system bus
AA = Intel® Pentium® processor Extreme Edition and Intel® Pentium® D processor on 65 nm process
AB = Intel® Pentium® 4 processor on 65 nm process
AC = Intel® Celeron® processor in 478-pin package
AD = Intel® Pentium® D processor on 65 nm process
AE = Intel® Core™ Duo processor and Intel® Core™ Solo processor on 65 nm process
AF = Dual-Core Intel® Xeon® processor LV

The Specification Updates for the Pentium® processor, Pentium® Pro processor, and other Intel products do not use this convention.

Errata (Sheet 1 of 2)

No.	C1	Plans	Description
I1	x	No Fix	Access to an Unsupported Address Range in Uniprocessor (UP) or Dual-processor (DP) Systems Supporting Intel® Virtualization Technology May Not Trigger Appropriate Actions.
I2	x	Plan Fix	VM Exit Due to a MOV from CR8 May Cause an Unexpected Memory Access
I3	x	No Fix	The Processor May Incorrectly Respond to Machine Checks during VM Entry/Exit Transitions
I4	x	Plan Fix	Power Down Requests May not be Serviced if a Power Down Transition is Interrupted by an In-Target Probe Event in the Presence of a Specific Type of VM Exit
I5	x	No Fix	Two Correctable L2 Cache Errors in Close Proximity May Cause a System Hang
I6	x	No Fix	Processor May Hang with a 25% or Less STPCLK# Duty Cycle
I7	x	No Fix	Writing the Local Vector Table (LVT) When an Interrupt is Pending May Cause an Unexpected Interrupt
I8	x	No Fix	The Execution of VMPTRLD or VMREAD May Cause an Unexpected Memory Access
I9	x	No Fix	IRET under Certain Conditions May Cause an Unexpected Alignment Check Exception
I10	x	No Fix	The Execution of a VMPTRLD Instruction May Cause an Unexpected Memory Access
I11	x	No Fix	IA32_THERM_STATUS MSR bits [5:4] Are Only Cleared when RESET# is Asserted
I12	x	No Fix	Control Register 2 (CR2) Can be Updated during a REP MOVS/STOS Instruction with Fast Strings Enabled
I13	x	No Fix	A 64-Bit Value of Linear Instruction Pointer (LIP) May be Reported Incorrectly in the Branch Trace Store (BTS) Memory Record or in the Precise Event Based Sampling (PEBS) Memory Record
I14	x	No Fix	A Push of ESP That Faults May Zero the Upper 32 Bits of RSP
I15	x	No Fix	BTS (Branch Trace Store) and PEBS (Precise Event Based Sampling) May Update Memory outside the BTS/PEBS Buffer
I16	x	No Fix	Data Breakpoints on the High Half of a Floating Point Line Split May Not Be Captured
I17	x	No Fix	MOV CR3 Performs Incorrect Reserved Bit Checking When in PAE Paging
I18	x	No Fix	Checking of Page Table Base Address May Not Match the Address bit Width Supported by the Platform.
I19	x	No Fix	With TF (Trap Flag) Asserted, FP Instruction That Triggers an Unmasked FP Exception May Take Single Step Trap before Retirement of Instruction
I20	x	No Fix	FXRSTOR May Not Restore Non-canonical Effective Addresses on Processors with Intel® Extended Memory 64 Technology (Intel® EM64T) Enabled.
I21	x	No Fix	Machine Check Exceptions May not Update Last-Exception Record MSRs (LERs)
I22	x	No Fix	Locks and SMC Detection May Cause the Processor to Temporarily Hang
I23	x	No Fix	REP STOS/MOVS Instructions with RCX>=2^32 May Cause a System Hang
I24	x	Plan Fix	VMEntry from 64-bit Host to 32-bit Guest may Cause IERR# with Hyper-Threading Technology Enabled

Errata (Sheet 2 of 2)

No.	C1	Plans	Description
I25	x	No Fix	The IA32_MCi_STATUS MSR May Improperly Indicate that Additional MCA Information May Have Been Captured
I26	x	No Fix	Memory Aliasing of Pages as Uncacheable Memory Type and Write Back (WB) May Hang the System
I27	x	Plan Fix	A VM Exit due to SMI or INIT in Parallel with a Pending FP Exception May Not Correctly Clear the Interruptibility State Bits
I28	x	Plan Fix	Attempting to Use an LDT Entry when the LDTR Has Been Loaded with an Unusable Segment May Cause Unexpected Memory Accesses
I29	x	Plan Fix	VM Entry/Exit Writes to LSTAR/SYSCALL_FLAG MSR's May Cause Incorrect Data to be Written to Bits [63:32]
I30	x	Plan Fix	At a Bus Ratio of 13:1, RCNT and Address Parity May be Incorrect
I31	x	Plan Fix	On a "Failed VM-entry" VM Exit, the VMCS Pointer May have Incorrect Value
I32	x	Fixed	During an Enhanced HALT or Enhanced Intel SpeedStep® Technology Ratio Transition the System May Hang
I33	x	Plan Fix	VMLAUNCH/VMRESUME May Not Fail when VMCS is Programmed to Cause VM Exit to Return to a Different Mode
I34	x	Plan Fix	NMI-blocking Information Recorded in VMCS May be Incorrect after a #GP on an IRET Instruction
I35	x	No Fix	FS/GS Base MSRs can be Loaded from MSR-Load Areas during VM Entry or VM Exit
I36	x	No Fix	Processor May Fault when the Upper 8 Bytes of Segment Selector is Loaded From a Far Jump Through a Call Gate via the Local Descriptor Table
I37	x	No Fix	L2 Cache ECC Machine Check Errors May be Erroneously Reported after an Asynchronous RESET# Assertion
I38	x	No Fix	The Processor May Issue Front Side Bus Transactions up to 6 Clocks after RESET# is Asserted
I39	x	Plan Fix	VM EXIT Due to TPR shadow Threshold May Improperly Set and Cause "Blocking by STI" actions
I40	x	No Fix	Processor May Hang During Entry into No-Fill Mode or No-Eviction Mode
I41	x	Plan Fix	VMCALL to Activate Dual-monitor Treatment of SMIs and SMM Ignores Reserved Bit settings in VM-exit Control Field
I42	x	No Fix	Using 2M/4M Pages When A20M# is Asserted May Result in Incorrect Address Translations
I43	x	No Fix	Writing Shared Unaligned Data that Crosses a Cache Line without Proper Semaphores or Barriers May Expose a Memory Ordering Issue
I44	x	No Fix	Front Side Bus Machine Checks May be Reported as a Result of On-Going Transactions during Warm Reset

Specification Changes

No.	SPECIFICATION CHANGES
	None for this revision of this specification update.

Specification Clarifications

No.	SPECIFICATION CLARIFICATIONS
	None for this revision of this specification update.

Documentation Changes

No.	DOCUMENTATION CHANGES
	None for this revision of this specification update.

Errata

I1. Access to an Unsupported Address Range in Uniprocessor (UP) or Dual-processor (DP) Systems Supporting Intel® Virtualization Technology May Not Trigger Appropriate Actions.

Problem: When using processors supporting Intel® Virtualization Technology and configured as dual- or single-processor-capable (that is, not multiprocessor-capable), the processor should perform address checks using a maximum physical address width of 36. Instead, these processors will perform address checks using a maximum physical address width of 40.

Implication: Due to this erratum, actions which are normally taken upon detection of an unsupported address may not occur. Software which does not attempt to access unsupported addresses will not experience this erratum.

Workaround: None identified.

Status: For the steppings affected, see the *Summary Table of Changes*.

I2. VM Exit Due to a MOV from CR8 May Cause an Unexpected Memory Access

Problem: In a system supporting Intel Virtualization Technology and Intel® EM64T, if the "CR8-store exiting" bit in the processor-based VM-execution control field is set and the "use TPR shadow" bit is not set, a MOV from CR8 instruction executed by a Virtual Machine Extensions (VMX) guest that causes a VM exit may generate an unexpected memory access.

Implication: When this erratum occurs, a read access to unexpected address may be issued to the chipset. Subsequent side effects are dependent on chipset operation and may include system hang.

Workaround: It is possible for the BIOS to contain a workaround for this erratum.

Status: For the steppings affected, see the *Summary Table of Changes*.

I3. The Processor May Incorrectly Respond to Machine Checks during VM Entry/Exit Transitions

Problem: In systems supporting Intel Virtualization Technology, when machine checks are encountered during VM entry/exit transitions, the processor is expected to respond with a VM exit (if a machine check occurs during VM entry) or abort (if a machine check occurs during VM exit). As a result of this erratum when machine checks occur during VM entry/exit transitions the processor will attempt to service the machine check which may lead to IERR-shutdown or execution of the Machine Check handler, dependent on the CR4.MCE setting.

Implication: The system may end up in the shutdown state if CR4.MCE is not set.

Workaround: No identified.

Status: For the steppings affected, see the *Summary Table of Changes*.

I4. Power Down Requests May not be Serviced if a Power Down Transition is Interrupted by an In-Target Probe Event in the Presence of a Specific Type of VM Exit

Problem: In a system supporting Intel Virtualization Technology, the processor may service a pended VM exit prior to completely exiting out of a low power state when the following sequences of events occur: 1) Chip-wide power down transition occurs and 2) VM exit due to a VMLaunch, VMResume, STI, POPF, POPFD, or IRET instruction is pended and 3) Chip-wide power down transition is interrupted by an In-Target Probe event.

Implication: Due to this erratum the processor may not recognize further STPCLK# assertions, TM1, TM2, or Enhanced Intel SpeedStep® Technology. Intel has not observed this erratum with any commercially available software.

Workaround: No identified.

Status: For the steppings affected, see the *Summary Table of Changes*.

I5. Two Correctable L2 Cache Errors in Close Proximity May Cause a System Hang

Problem: If two correctable L2 cache errors are detected in close proximity to each other, a livelock may occur as a result of the processor being unable to resolve this condition.

Implication: When this erratum occurs, the processor may livelock and result in a system hang. Intel has only observed this erratum while injecting cache errors in simulation.

Workaround: No identified.

Status: For the steppings affected, see the *Summary Table of Changes*.

I6. Processor May Hang with a 25% or Less STPCLK# Duty Cycle

Problem: If a system de-asserts STPCLK# at a 25% or less duty cycle and the processor thermal control circuit (TCC) on-demand clock modulation is active, the processor may hang. This erratum does not occur under the automatic mode of the TCC.

Implication: When this erratum occurs, the processor may hang.

Workaround: If use of the on-demand mode of the processor's TCC is desired in conjunction with STPCLK# modulation, then assure that STPCLK# is not asserted at a 25% duty cycle.

Status: For the steppings affected, see the *Summary Table of Changes*.

I7. Writing the Local Vector Table (LVT) When an Interrupt is Pending May Cause an Unexpected Interrupt

Problem: If a local interrupt is pending when the LVT entry is written, an interrupt may be taken on the new interrupt vector even if the mask bit is set.

Implication: An interrupt may immediately be generated with the new vector when a LVT entry is written, even if the new LVT entry has the mask bit set. If there is no Interrupt Service Routine (ISR) set up for that vector the system will GP fault. If the ISR does not do an End of Interrupt (EOI) the bit for the vector will be left set in the in-service register and mask all interrupts at the same or lower priority.

Workaround: Any vector programmed into an LVT entry must have an ISR associated with it, even if that vector was programmed as masked. This ISR routine must do an EOI to clear any unexpected interrupts that may occur. The ISR associated with the spurious vector does not generate an EOI, therefore the spurious vector should not be used when writing the LVT.

Status: For the steppings affected, see the *Summary Table of Changes.*

I8. The Execution of VMPTRLD or VMREAD May Cause an Unexpected Memory Access

Problem: On processors supporting Intel Virtualization Technology, executing a VMPTRLD or a VMREAD instruction outside of VMX mode may result in a load to an unexpected address.

Implication: This erratum may cause a load to an unexpected memory address.

Workaround: It is possible for the BIOS to contain a workaround for this erratum.

Status: For the steppings affected, see the *Summary Table of Changes.*

I9. IRET under Certain Conditions May Cause an Unexpected Alignment Check Exception

Problem: In IA-32e mode, it is possible to get an Alignment Check Exception (#AC) on the IRET instruction even though alignment checks were disabled at the start of the IRET. This can only occur if the IRET instruction is returning from CPL3 code to CPL3 code. IRETs from CPL0/1/2 are not affected. This erratum can occur if the EFLAGS value on the stack has the AC flag set, and the interrupt handler's stack is misaligned. In IA-32e mode, RSP is aligned to a 16-byte boundary before pushing the stack frame.

Implication: In IA-32e mode, under the conditions given above, an IRET can get a #AC even if alignment checks are disabled at the start of the IRET. This erratum can only be observed with a software generated stack frame.

Workaround: Software should not generate misaligned stack frames for use with IRET.

Status: For the steppings affected, see the *Summary Table of Changes.*

I10. The Execution of a VMPTRLD Instruction May Cause an Unexpected Memory Access

Problem: In a system supporting Intel Virtualization Technology, executing VMPTRLD may cause a memory access to an address not referenced by the memory operand.

Implication: This erratum may cause unpredictable system behavior including system hang.

Workaround: It is possible for the BIOS to contain a workaround for this erratum.

Status: For the steppings affected, see the *Summary Table of Changes.*

I11. IA32_THERM_STATUS MSR bits [5:4] Are Only Cleared when RESET# is Asserted

Problem: Bits [5:4] of the IA32_THERM_STATUS MSR are only cleared when RESET# is asserted: 1) Bit 5 {Out of Spec Temperature Status Log} can not be reset by software. Once bit 5 is set due to being above the maximum operating temperature, it will remain 1 until the next assertion of RESET#. Software writing a 0 to this register will not clear the bit. 2) Bit 4 {Out of Spec Temperature Status} does not reflect the correct status after the processor returns below the maximum operating temperature. Bit 4 will remain 1 until the next assertion of RESET#.

Implication: Bits [5:4] of the affected MSR will be 1 after PROCHOT# is asserted, and they can not be cleared.

Workaround: None identified.

Status: For the steppings affected, see the *Summary Table of Changes*.

I12. Control Register 2 (CR2) Can be Updated during a REP MOVS/STOS Instruction with Fast Strings Enabled

Problem: Under limited circumstances while executing a REP MOVS/STOS string instruction, with fast string enabled, it is possible for the value in CR2 to be changed as a result of an interim paging event, normally invisible to the user. Any higher priority architectural event that arrives and is handled while the interim paging event is occurring may see the modified value of CR2.

Implication: The value of CR2 is correct at the time that an architectural page fault is signaled. Intel has not observed this erratum with any commercially available software.

Workaround: None identified.

Status: For the steppings affected, see the *Summary Table of Changes*.

I13. A 64-Bit Value of Linear Instruction Pointer (LIP) May be Reported Incorrectly in the Branch Trace Store (BTS) Memory Record or in the Precise Event Based Sampling (PEBS) Memory Record

Problem: On a processor supporting Intel EM64T, 1) If an instruction fetch wraps around the 4G boundary in Compatibility Mode, the 64-bit value of LIP in the BTS memory record will be incorrect (upper 32 bits will be set to FFFFFFFFh when they should be 0). 2) If a PEBS event occurs on an instruction whose last byte is at memory location FFFFFFFFh, the 64-bit value of LIP in the PEBS record will be incorrect (upper 32 bits will be set to FFFFFFFFh when they should be 0).

Implication: Intel has not observed this erratum on any commercially available software.

Workaround: None identified.

Status: For the steppings affected, see the *Summary Table of Changes*.

I14. A Push of ESP That Faults May Zero the Upper 32 Bits of RSP

Problem: In the event that a push ESP instruction, that faults, is executed in compatibility mode, the processor will incorrectly zero upper 32-bits of RSP.

Implication: A Push of ESP in compatibility mode will zero the upper 32-bits of RSP. Due to this erratum, this instruction fault may change the contents of RSP. This erratum has not been observed in commercially available software.

Workaround: None identified.

Status: For the steppings affected, see the *Summary Table of Changes*.

I15. BTS (Branch Trace Store) and PEBS (Precise Event Based Sampling) May Update Memory outside the BTS/PEBS Buffer

Problem: If the BTS/PEBS if defined such that: 1) The difference between BTS/PEBS buffer base and BTS/PEBS absolute maximum is not an integer multiple of the corresponding record sizes; 2) BTS/PEBS absolute maximum is less than a record size from the end of the virtual address space; 3) The record that would cross BTS/PEBS absolute maximum will also continue past the end of the virtual address space; A BTS/PEBS record can be written that will wrap at the 4G boundary (IA32) or 2^64 boundary (EM64T mode), and write memory outside of the BTS/PEBS buffer.

Implication: Software that uses BTS/PEBS near the 4G boundary (IA32) or 2^64 boundary (EM64T mode), and defines the buffer such that it does not hold an integer multiple of records can update memory outside the BTS/PEBS buffer.

Workaround: Define BTS/PEBS buffer such that BTS/PEBS absolute maximum minus BTS/PEBS buffer base is integer multiple of the corresponding record sizes as recommended in the IA-32 Intel® Architecture Software Developer's Manual, Volume 3.

Status: For the steppings affected, see the *Summary Table of Changes*.

I16. Data Breakpoints on the High Half of a Floating Point Line Split May Not Be Captured

Problem: When a floating point load which splits a 64-byte cache line gets a floating point stack fault, and a data breakpoint register maps to the high line of the floating point load, internal boundary conditions exist that may prevent the data breakpoint from being captured.

Implication: When this erratum occurs, a data breakpoint will not be captured.

Workaround: None identified.

Status: For the steppings affected, see the *Summary Table of Changes*.

I17. MOV CR3 Performs Incorrect Reserved Bit Checking When in PAE Paging

Problem: The MOV CR3 instruction should perform reserved bit checking on the upper unimplemented address bits. This checking range should match the address width reported by CPUID instruction 0x8000008. This erratum applies whenever PAE is enabled.

Implication: Software that sets the upper address bits on a MOV CR3 instruction and expects a fault may fail. This erratum has not been observed with commercially available software.

Workaround: None identified.

Status: For the steppings affected, see the *Summary Table of Changes*.

I18. Checking of Page Table Base Address May Not Match the Address Bit Width Supported by the Platform

Problem: If the page table base address, included in the page map level-4 table, page-directory pointer table, page-directory table or page table, exceeds the physical address range supported by the platform (for example, 36-bit) and it is less than the implemented address range (for example, 40-bit), the processor does not check if the address is invalid.

Implication: If software sets such invalid physical address in those tables, the processor does not generate a page fault (#PF) upon access to that virtual address, and the access results in an incorrect read or write. If BIOS provides only valid physical address ranges to the operating system, this erratum will not occur.

Workaround: BIOS must provide valid physical address ranges to the operating system.

Status: For the steppings affected, see the *Summary Table of Changes*.

I19. With TF (Trap Flag) Asserted, FP Instruction That Triggers an Unmasked FP Exception May Take Single Step Trap before Retirement of Instruction

Problem: If an FP instruction generates an unmasked exception with the EFLAGS.TF=1, it is possible for external events to occur, including a transition to a lower power state. When resuming from the lower power state, if may be possible to take the single step trap before the execution of the original FP instruction completes.

Implication: A Single Step trap will be taken when not expected.

Workaround: None identified.

Status: For the steppings affected, see the *Summary Table of Changes.*

I20. FXRSTOR May Not Restore Non-canonical Effective Addresses on Processors with Intel® Extended Memory 64 Technology (Intel® EM64T) Enabled

Problem: If an x87 data instruction has been executed with a non-canonical effective address, FXSAVE may store that non-canonical FP Data Pointer (FDP) value into the save image. An FXRSTOR instruction executed with 64-bit operand size may signal a General Protection Fault (#GP) if the FDP or FP instruction Pointer (FIP) is in non-canonical form.

Implication: When this erratum occurs, Intel EM64T enabled systems may encounter an unintended #GP fault.

Workaround: Software should avoid using non-canonical effective addressing in EM64T enabled processors. BIOS can contain a workaround for this erratum removing the unintended #GP fault on FXRSTOR.

Status: For the steppings affected, see the *Summary Table of Changes.*

I21. Machine Check Exceptions May not Update Last-Exception Record MSRs (LERs)

Problem: The Last-Exception Record MSRs (LERs) may not get updated when Machine Check Exceptions occur

Implication: When this erratum occurs, the LER may not contain information relating to the machine check exception. They will contain information relating to the exception prior to the machine check exception.

Workaround: None identified.

Status: For the steppings affected, see the *Summary Table of Changes.*

I22. Bus Locks and SMC Detection May Cause the Processor to Hang Temporarily

Problem: The processor may temporarily hang in an HT Technology enabled system, if one logical processor executes a synchronization loop that includes one or more locks and is waiting for release by the other logical processor. If the releasing logical processor is executing instructions that are within the detection range of the self-modifying code (SMC) logic, then the processor may be locked in the synchronization loop until the arrival of an interrupt or other event.

Implication: If this erratum occurs in an HT Technology enabled system, the application may temporarily stop making forward progress. Intel has not observed this erratum with any commercially available software.

Workaround: None identified.

Status: For the steppings affected, see the *Summary Table of Changes.*

I23. REP STOS/MOVS Instructions with RCX>=2^32 May Cause a System Hang

Problem: In IA-32e mode using Intel EM64T-enabled processors, executing a repeating string instruction with the iteration count greater than or equal to 2^32 and a pending event may cause the REP STOS/MOVS instruction to live lock and hang.

Implication: When this erratum occurs, the processor may live lock and result in a system hang. Intel has not observed this erratum with any commercially available software.

Workaround: Do not use strings larger than 4 GB.

Status: For the steppings affected, see the *Summary Table of Changes.*

I24. VMEntry from 64-bit Host to 32-bit Guest may Cause IERR# with Hyper-Threading Technology Enabled

Problem: When transitioning from a 64-bit host environment to a 32-bit guest environment via a VMEntry, internal conditions in a processor with Hyper-Threading enabled may cause a page-table walk to be prematurely terminated, resulting in a processor hang and the assertion of IERR#.

Implication: An IERR# may occur on VMEntry from a 64-bit to a 32-bit environment with Hyper-Threading Technology enabled.

Workaround: It is possible for the BIOS to contain a workaround for this erratum.

Status: For the steppings affected, see the *Summary Table of Changes.*

I25. The IA32_MCi_STATUS MSR May Improperly Indicate that Additional MCA Information May Have Been Captured

Problem: When a data parity error is detected and the bus queue is busy, the ADDRV and MISCV bits of the IA32_MCi_STATUS register may be asserted even though the contents of the IA32_MCi_ADDR and IA32_MCi_MISC MSRs were not properly captured.

Implication: If this erratum occurs, the MCA information captured in the IA32_MCi_ADDR and IA32_MCi_MISC may not respond to the reported machine-check error, even though the ADDRV and MISCV are asserted.

Workaround: None identified.

Status: For the steppings affected, see the *Summary Table of Changes.*

I26. Memory Aliasing of Pages As Uncacheable Memory Type and Write Back (WB) May Hang the System

Problem: When a page is being accessed as either Uncacheable (UC) or Write Combining (WC) and Write Back (WB), under certain bus and memory timing conditions, the system may loop in a continual sequence of UC fetch, implicit writeback, and Request for Ownership (RFO) retries.

Implication: This erratum has not been observed in any commercially available operating system or application. The aliasing of memory regions, a condition necessary for this erratum to occur, is documented as being unsupported in the *IA-32 Intel® Architecture Software Developer's Manual*, Volume 3, section 10.12.4, Programming the PAT. However, if this erratum occurs the system may hang.

Workaround: None identified.

Status: For the steppings affected, see the *Summary Table of Changes.*

I27. A VM Exit due to SMI or INIT in Parallel with a Pending FP Exception May Not Correctly Clear the Interruptibility State Bits

Problem: When a pending FP exception is ready to be taken, a VM exit due to SMI or INIT may not clear Blocking by STI and/or Blocking by MOV SS bits correctly in Virtual-Machine Control Structure (VMCS) as expected.

Implication: A VM exit due to SMI or INIT may show incorrect STI and/or MOV SS blocking state in VM-exit Interruptibility field.

Workaround: It is possible for the BIOS to contain a workaround for this erratum.

Status: For the steppings affected, see the *Summary Table of Changes.*

I28. Attempting to Use an LDT Entry when the LDTR Has Been Loaded with an Unusable Segment May Cause Unexpected Memory Accesses

Problem: In a system supporting Virtualization Technology, the processor may incorrectly VM exit under the following conditions: 1) Interrupt-Window-Exiting VM-execution control is set; 2) RFLAGS [IF]=1; 3) Chipwide Powerdown transition requests occur when the processor is in Wait-for-SIPI or Shutdown states.

Implication: Due to this erratum, Interrupt-Window-Exiting VM exits may take the logical processor out of Wait-For-SIPI and Shutdown states. Intel has not observed this erratum with any commercially available software.

Workaround: It is possible for the BIOS to contain a workaround for this erratum.

Status: For the steppings affected, see the *Summary Table of Changes.*

I29. VM Entry/Exit Writes to LSTAR/SYSCALL_FLAG MSR's May Cause Incorrect Data to be Written to Bits [63:32]

Problem: Incorrect MSR data in bits [63:32] may be observed in the following two cases: 1) When ECX contains 0xC0000082 bits [63:32] of the data may be zeroed. 2) When ECX does not contain 0xC0000084 and a VM entry/exit writes the IA32_CR_SYSCALL_FLAG_MASK MSR (MSR Address 0xC0000084) bits [63:32] of the data may not be zeroed.

Implication: Bits [63:32] of the affected MSRs may contain the wrong data after a VM exit/entry which loads the affected MSR.

Workaround: It is possible for the BIOS to contain a workaround for this erratum.

Status: For the steppings affected, see the *Summary Table of Changes.*

I30. At a Bus Ratio of 13:1, RCNT and Address Parity May be Incorrect

Problem: In a system running at 13:1 core-to-bus ratio, RCNT[0] (ADDR#[28], phase b) may report incorrect information.

Implication: RCNT[0] may contain incorrect information and cause address parity machine check errors.

Workaround: Address parity should be disabled and RCNT information should be ignored at the bus ratio of 13:1.

Status: For the steppings affected, see the *Summary Table of Changes.*

I31. On a “Failed VM-entry” VM Exit, the VMCS Pointer May have Incorrect Value

Problem: On a “failed VM-entry” VM exit, the VMCS pointer may have incorrect value.

Implication: The value of the VMCS pointer may be incorrect and may result in unpredictable behavior after the “failed VM-entry”.

Workaround: It is possible for the BIOS to contain a workaround for this erratum.

Status: For the steppings affected, see the *Summary Table of Changes.*

I32. VMLAUNCH/VMRESUME May Not Fail when VMCS is Programmed to Cause VM Exit to Return to a Different Mode

Problem: VMLAUNCH/VMRESUME instructions may not fail if the value of the “host address-space size” VM-exit control differs from the setting of IA32_EFER.LMA.

Implication: Programming the VMCS to allow the monitor to be in different modes prior to VMLAUNCH/VMRESUME and after VM-exit may result in undefined behavior.

Workaround: Software should ensure that "host address-space size" VM-exit control has the same value as IA32_EFER.LMA at the time of VMLAUNCH/VMRESUME.

Status: For the steppings affected, see the *Summary Table of Changes.*

I33. NMI-blocking Information Recorded in VMCS May be Incorrect after a #GP on an IRET Instruction

Problem: In a system supporting Intel Virtualization Technology, the NMI blocking bit in the Interruption-Information Field in the guest VMCS may be set incorrectly. This erratum will happen if a VMExit occurs for a #GP fault on an IRET instruction due to an EIP that violates the segment limit or is non-canonical.

Implication: If this erratum occurs, monitor software may not be able to handle #GP and then inject an NMI since monitor software does not have information about whether NMIs are blocked in the guest.

Workaround: Monitor software can workaround this bug by avoiding injection of NMI after #GP emulation.

Status: For the steppings affected, see the *Summary Table of Changes.*

I34. FS/GS Base MSRs can be Loaded from MSR-Load Areas during VM Entry or VM Exit

Problem: If the VM Exit or VM Entry MSR load area contains references to the FS or GS Base MSRs, the VM Exit and VM Entry transitions should fail. Instead, the operation will load the MSRs with the value in the corresponding MSR-load area entry.

Implication: VM Entries and VM Exits that should fail will complete successfully in this situation. If a VM entry is to virtual-8086 mode, the base address for FS or for GS may be loaded with a value that is not consistent with that mode. Intel has not observed this erratum with any commercially available software or systems.

Workaround: Software should not enter values in the MSR-load areas that correspond to either the FS Base MSR or the GS Base MSR. Software can establish the value of these registers on VM entry using the guest-state area of the Virtual-Machine Control Structure (VMCS) and on VM exit using the host-state area of the VMCS.

Status: For the steppings affected, see the *Summary Table of Changes.*

I35. Processor May Fault when the Upper 8 Bytes of Segment Selector is Loaded From a Far Jump Through a Call Gate via the Local Descriptor Table

Problem: In IA-32e mode of the Intel EM64T processor, control transfers through a call gate via the Local Descriptor Table (LDT) that uses a 16-byte descriptor, the upper 8-byte access may wrap and access an incorrect descriptor in the LDT. This only occurs on an LDT with a LIMIT>0x10008 with a 16-byte descriptor that has a selector of 0xFFFC.

Implication: In the event this erratum occurs, the upper 8-byte access may wrap and access an incorrect descriptor within the LDT, potentially resulting in a fault or system hang. Intel has not observed this erratum with any commercially available software.

Workaround: None identified.

Status: For the steppings affected, see the *Summary Table of Changes.*

I36. L2 Cache ECC Machine Check Errors May be Erroneously Reported after an Asynchronous RESET# Assertion

Problem: Machine check status MSRs may incorrectly report the following L2 Cache ECC machine-check errors when cache transactions are in-flight and RESET# is asserted: 1) Instruction Fetch Errors (IA32_MC2_STATUS with MCA error code 153); 2) L2 Data Write Errors (IA32_MC1_STATUS with MCA error code 145).

Implication: Uncorrected or corrected L2 ECC machine check errors may be erroneously reported. Intel has not observed this erratum on any commercially available system.

Workaround: When a real run-time L2 Cache ECC Machine Check occurs, a corresponding valid error will normally be logged in the IA32_MC0_STATUS register. BIOS may clear IA32_MC2_STATUS and/or IA32_MC1_STATUS for these specific errors when IA32_MC0_STATUS does not have its VAL flag set.

Status: For the steppings affected, see the *Summary Table of Changes.*

I37. The Processor May Issue Front Side Bus Transactions up to 6 Clocks after RESET# is Asserted

Problem: The processor may issue transactions beyond the documented 3 Front Side Bus (FSB) clocks and up to 6 FSB clocks after RESET# is asserted in the case of a warm reset. A warm reset is where the chipset asserts RESET# when the system is running.

Implication: The processor may issue transactions up to 6 FSB clocks after the RESET# is asserted.

Workaround: None identified.

Status: For the steppings affected, see the *Summary Table of Changes.*

I38. VM Exit Due to TPR Shadow Below Threshold May Improperly Set and Cause “Blocking by STI” actions

Problem: In a system supporting Intel Virtualization Technology and Intel EM64T, the “blocking by STI” bit of the interruptibility-state field may be saved as 1 rather than 0. This erratum may occur when a STI instruction is executed directly prior to a MOV to CR8 which results in a VM exit due to a reduction of the TPR shadow value below the TPR threshold.

Implication: When this erratum occurs, delivery of an interrupt may be delayed by one instruction.

Workaround: It is possible for the BIOS to contain a workaround for this erratum.

Status: For the steppings affected, see the *Summary Table of Changes.*

I39. Processor May Hang During Entry into No-Fill Mode or No-Eviction Mode

Problem: Only one logical processor per core can be active when processor is put in No-Fill Mode or No-Eviction Mode. If the other logical processor is active or there is an internal or external event pending to wake that logical processor, the processor may hang when writing to MSR IA32_BIOS_CACHE_AS_RAM (80H).

Implication: A processor may hang due to this erratum. Intel has not observed this erratum with any commercially available software or system.

Workaround: None identified.

Status: For the steppings affected, see the *Summary Table of Changes.*

I40. VMCALL to Activate Dual-monitor Treatment of SMIs and SMM Ignores Reserved Bit Settings in VM-exit Control Field

Problem: Processors supporting Intel Virtualization Technology can execute VMCALL from within the Virtual Machine Monitor (VMM) to activate dual-monitor treatment of SMIs and SMM. Due to this erratum, if reserved bits are set to values inconsistent with VMX Capability MSRs, VMCALL may not VMFail.

Implication: VMCALL executed to activate dual-monitor treatment of SMIs and SMM may not VMFail due to incorrect reserved bit settings in VM-Exit control field.

Workaround: Software should ensure that all VMCS reserved bits are set to values consistent within VMX Capability MSRs.

Status: For the steppings affected, see the *Summary Table of Changes.*

I41. Using 2M/4M Pages when A20M# Is Asserted May Result in Incorrect Address Translations

Problem: An external A20M# pin, if enabled, forces address bit 20 to be masked (forced to zero) to emulates real-address mode address wraparound at 1 megabyte. However, if all of the following conditions are met, address bit 20 may not be masked: 1) paging is enabled; 2) a linear address has bit 20 set; 3) the address references a large page; or 4) A20M# is enabled.

Implication: When A20M# is enabled and an address references a large page the resulting translated physical address may be incorrect. This erratum has not been observed with any commercially available operating system.

Workaround: Operating systems should not allow A20M# to be enabled if the masking of address bit 20 could be applied to an address that references a large part. A20M# is normally only used with the first megabyte of memory.

Status: For the steppings affected, see the *Summary Table of Changes.*

I42. Writing Shared Unaligned Data that Crosses a Cache Line without Proper Semaphores or Barriers May Expose a Memory Ordering Issue

Problem: Software which is written so that multiple agents can modify the same shared unaligned memory location at the same time may experience a memory ordering issue if multiple loads access this shared data shortly thereafter. Exposure to this problem requires the use of a data write which spans a cache line boundary.

Implication: This erratum may cause loads to be observed out of order. Intel has not observed this erratum with any commercially available software or system.

Workaround: Software should ensure at least one of the following is true when modifying shared data by multiple agents: 1) The shared data is aligned; or 2) Proper semaphores or barriers are used in order to prevent concurrent data accesses.

Status: For the steppings affected, see the *Summary Table of Changes.*

I43. Front Side Bus Machine Checks May be Reported as a Result of On-Going Transactions during Warm Reset

Problem: Processor Front Side Bus (FSB) protocol/signal integrity machine checks may be reported if the transactions are initiated or in-progress during a warm reset. A warm reset is where the chipset asserts RESET# when the system is running.

Implication: The processor may log FSB protocol/signal integrity machine checks if transactions are allowed to occur during RESET# assertions.

Workaround: BIOS may clear FSB protocol/signal integrity machine checks for systems/chipsets which do not block new transactions during RESET# assertions.

Status: For the steppings affected, see the *Summary Table of Changes.*

I44. The IA32_MC0_STATUS and IA32_MC1_STATUS Overflow Bit is not set when Multiple Un-correctable Machine Check Errors Occur at the Same Time

Problem: When two MC0/MC1 enabled un-correctable machine check errors are detected in the same internal clock cycle, the highest priority error will be logged in IA32_MC0_STATUS / IA32_MC1_STATUS register, but the overflow bit may not be set.

Implication: The highest priority error will be logged and signaled if enabled, but the overflow bit in the IA32_MC0_STATUS/ IA32_MC1_STATUS register may not be set.

Workaround: None identified.

Status: For the steppings affected, see the *Summary Table of Changes.*

Specification Changes

There are no new Specification Changes for this revision.

The Specification Changes listed in this section apply to the following documents:

1. *Dual-Core Intel® Xeon® Processor 5000 Series Datasheet* (Document Number 313079)

All Specification Changes will be incorporated into a future version of the appropriate Intel® Xeon® processor documentation.

Specification Clarifications

There are no new Specification Clarifications for this revision.

The Specification Clarifications listed in this section apply to the following documents:

1. *Dual-Core Intel® Xeon® Processor 5000 Series* Datasheet (Document Number 313079)

All Specification Changes will be incorporated into a future version of the appropriate Intel® Xeon® processor documentation.

Documentation Changes

Note: Documentation changes for *IA-32 Intel® Architecture Software Developer's Manual* volumes 1, 2A, 2B, 3A and 3B will be posted in the separate document *IA-32 Intel® Architecture Software Developer's Manual Documentation Changes*. Follow the link below to become familiar with this file.

http://developer.intel.com/design/pentium4/specupdt/252046.htm

There are no new Documentation Changes for this revision.

The Documentation Changes listed in this section apply to the following documents:

1. *Dual-Core Intel® Xeon® Processor 5000 Series Datasheet* (Document Number 313079)

All Documentation Changes will be incorporated into a future version of the appropriate Intel® Xeon® processor documentation.